AMENDMENTS TO THE CLAIMS

Please accept amended claim 1 as follows:

1. (Currently Amended) A communication system which stores packet data received via a plurality of channels in a memory or transmits packet data stored in a memory through the plurality of communication channels, the communication system comprising:

a plurality of buffer descriptors in which information on packet data received or transmitted via the plurality of communication channels is stored;

a central processing unit (CPU) which stores the information on packet data in each of the plurality of buffer descriptors, and allots a flag bit to each buffer descriptor indicating whether a buffer descriptor is being organized, whether an error occurred in packet data received via the plurality of communication channels, or whether the organization of each of the buffer descriptors is completed; and

a direct memory access (DMA) controller which determines the flag bit allotted by the central processing unit, and according to the flag bit, stops processing a buffer descriptor currently being accessed and accesses a next buffer descriptor, or processes packet data according to information ~~is~~ stored in the buffer descriptor currently being accessed.

2. (Original) The communication system of claim 1, wherein the flag bit comprises:

an ownership bit for indicating that the buffer descriptor is in a CPU mode if each of the buffer descriptors is being organized or an error occurred in packet data received via the communication channels, and for indicating that the buffer descriptor is in a DMA mode, in which the DMA controller is accessible, if the organization of each of the buffer descriptors is completed; and

a skip bit for indicating whether the CPU is organizing the buffer descriptor or whether or not an error occurred in packet data received via the communication channels.

3. (Original) The communication system of claim 1, wherein the CPU allots a next buffer descriptor pointer to each of the buffer descriptors, and the DMA controller accesses the next buffer descriptor based on an identification of the next buffer descriptor pointer allotted to the buffer descriptor currently being accessed.

4. (Original) The communication system of claim 1, wherein if processing of the buffer descriptor currently being accessed is completed, the DMA controller updates a value of a current pointer for indicating which buffer descriptor the DMA controller is currently accessing by adding an address corresponding to the size of a buffer descriptor to the value of the current pointer.

5. (Original) The communication system of claim 1, wherein the DMA controller comprises:

a start pointer for indicating a buffer descriptor to be accessed first; and

an address counter, which is initialized to the value of the start pointer, for counting the start address of the buffer descriptor to be accessed next, wherein the address counter updates the value of the address counter by adding an address corresponding to the size of a unit buffer descriptor to the value of the address counter.

6. (Original) A communication method performed in a communication system, which includes a CPU which stores information on packet data received/transmitted via a plurality of

communication channels in each buffer descriptor and includes a DMA controller which stores packet data received via the plurality of communication channels in a memory or transmits packet data stored in the memory via the plurality of communication channels, the communication method comprising:

(a) displaying a current status of a buffer descriptor using a flag bit, wherein the flag bit is allotted to the buffer descriptor by the CPU;

(b) determining whether the CPU is organizing a buffer descriptor to be processed by the DMA controller, whether error occurred in the packet data transmitted via the communication channels, or whether the DMA controller is accessible after the organization of the buffer descriptor is completed, by using the DMA controller to identify the flag bit;

(c) continuously identifying the flag bit using the DMA controller until the organization of the buffer descriptor to be accessed is completed if the CPU is organizing the buffer descriptor to be accessed in step (b);

(d) skipping the buffer descriptor currently being accessed if it is determined that error occurred in the communication channels in step (b), and performing step (b) again to process the next buffer descriptor;

(e) processing packet data whose information is stored in the buffer descriptor currently being accessed if the buffer descriptor currently being accessed is in a DMA mode, in which the DMA controller can access the buffer descriptor, in step (b); and

(f) converting the buffer descriptor processed in step (e) from the DMA mode into a CPU mode in which the CPU can access the buffer descriptor, and moving to a buffer descriptor to be processed next.

7. (Original) The communication method of claim 6, wherein the flag bit comprises:

an ownership bit for indicating that the buffer descriptor is in the CPU mode if the buffer descriptor is being organized by the CPU or an error occurred in packet data received via the communication channels, and for indicating that the buffer descriptor is in the DMA mode, in which the DMA controller is accessible, if the organization of the buffer descriptor is completed; and

a skip bit for indicating whether the CPU is organizing each of the buffer descriptor or whether or not an error occurred in packet data received via the communication channels.

8. (Original) The communication method of claim 7, wherein step (b) comprises:

(b1) identifying whether the buffer descriptor to be processed is in the CPU mode or the DMA mode with the DMA controller using the ownership bit;

(b2) performing step (e) if the buffer descriptor to be processed is determined to be in the DMA mode and identifying the skip bit if the buffer descriptor to be processed is determined to be in the DMA mode; and

(b3) performing step (d) if it is determined that error occurred in the communication channels based on the identification of the skip bit and performing step (c) if it is determined that the CPU is organizing the buffer descriptor to be processed based on the identification of the skip bit.

9. (Original) The communication method of claim 6, wherein the CPU allots a next buffer descriptor pointer to each of the buffer descriptors, and the DMA controller accesses the buffer

descriptor to be processed next based on the identification of the next buffer descriptor pointer allotted to each of the buffer descriptor.

10. (Original) The communication method of claim 6, wherein if processing the buffer descriptor currently being accessed is completed, the DMA controller updates the value of a current pointer for indicating which buffer descriptor the DMA controller is currently accessing by adding an address corresponding to the size of a unit buffer descriptor to the value of the current pointer.

11. (Original) The communication method of claim 6, further comprising the steps of:

indicating a buffer descriptor to be accessed first according to a start pointer; and

initializing an address counter to the value of the start pointer, for counting the start address of the buffer descriptor, wherein the address counter updates the value of the address counter by adding an address corresponding to the size of a unit buffer descriptor to the value of the address counter.

12. (Original) A buffer descriptor for storing packet data information comprising:

an ownership bit for indicating that the buffer descriptor is in a central processing unit (CPU) mode the buffer descriptor is being organized or an error occurred in packet data, and for indicating that the buffer descriptor is in a direct memory access (DMA) mode, in which a DMA controller is accessible, if the organization of the buffer descriptors is completed;

a skip bit for indicating whether a CPU is organizing the buffer descriptor or whether an error occurred in packet data received via a communication channel;

a status bit for indicating a status after a communication system transmits/receives packet data;

a data pointer for indicating an address in a memory in which packet data is stored or the address in the memory at which packet data to be transmitted through the communication channel is recorded; and

a command bit for indicating a packet data processing command.

13. (Original) The buffer descriptor of claim 12, wherein the status bit indicates a type of error.

14. (Original) The buffer descriptor of claim 12, further comprising a next buffer descriptor pointer (NBDP) for indicating a next buffer descriptor to be accessed the DMA controller.